ASSIGNMENT - III

1) Two nucleotide sequences:

GGCTGCAACTAGCTC

GGGTAAGCTTGC

Transition-Transversion scoring matrix:

	A	C	G	T
A	4	-1	1	-1
C	-1	4	-1	1
G	1	-1	4	-1
T	-1	1	-1	4

Gap penalty: -3

Global, local alignment (DP algorithm) and show final similarity score and the best alignment

Global alignment (using Needleman–Wunsch Algorithm)

		G	G	G	T	A	A	G	C	T	T	G	C
	0	-3	-6	-9	-12	-15	-18	-21	-24	-27	-30	-33	-36
G	-3	4	1	-2	-5	-8	-11	-14	-17	-20	-23	-26	-29
G	-6	1	8	5	2	-1	-4	-7	-10	-13	-16	-19	-22
C	-9	-2	5	7	6	3	0	-3	-3	-6	-9	-12	-15
T	-12	-5	2	4	11	8	5	2	-1	1	-2	-5	-8
G	-15	-8	-1	6	8	12	9	9	6	3	0	2	-1
C	-18	-11	-4	3	7	9	11	8	13	10	7	4	6
A	-21	-14	-7	0	4	11	13	12	10	12	9	8	5
A	-24	-17	-10	-3	1	8	15	14	11	9	11	10	7
C	-27	-20	-13	-6	-2	5	12	14	18	15	12	10	14
T	-30	-23	-16	-9	-2	2	9	11	15	22	19	16	13
A	-33	-26	-19	-12	-5	2	6	10	12	19	21	20	17
G	-36	-29	-22	-15	-8	-1	3	10	9	16	18	25	22
C	-39	-32	-25	-18	-11	-4	0	7	14	13	17	22	29
T	-42	-35	-28	-21	-14	-7	-3	4	11	18	17	19	26
C	-45	-38	-31	-24	-17	-10	-6	1	8	15	19	16	23

$d = -3$ Boundary conditions: $F(i,0) = F(0,i) = -id$

$$F(i,j) = \max \begin{cases} F(i-1, j-1) + s(x_i, y_j) \\ F(i-1, j) - d \\ F(i, j-1) - d \end{cases}$$

Final similarity score : 23

Best alignment :

GGCTGCAACTAGCTC
GGGTA_AGCTTG--C

---

local alignment (using Smith Waterman algorithm)

		G	G	G	T	A	A	G	C	T	T	G	C
	0	0	0	0	0	0	0	0	0	0	0	0	0
G	0	4	4	4	1	1	1	4	1	0	0	4	1
G	0	4	8	8	5	2	2	5	3	0	0	4	3
C	0	1	5	7	9	6	3	2	9	6	3	1	8
T	0	0	2	4	11	8	5	2	6	13	10	7	5
G	0	4	4	6	8	12	9	9	6	10	12	14	11
C	0	1	3	3	7	9	11	8	13	10	11	11	18
A	0	1	2	4	4	11	13	12	10	12	9	12	15
A	0	1	2	3	3	8	15	14	11	9	11	10	12
C	0	0	0	1	4	5	12	14	18	15	12	10	14
T	0	0	0	0	5	3	9	11	15	22	19	16	13
A	0	1	1	1	2	9	7	10	12	19	21	20	17
G	0	4	5	5	2	6	10	11	9	16	18	25	22
C	0	1	3	4	6	3	7	9	15	13	17	22	29
T	0	0	0	2	8	5	4	6	12	19	17	19	26
C	0	0	0	0	5	7	4	3	10	16	20	17	23

$d = -3$ Boundary conditions: $F(i,0) = F(0,i) = 0$

$$F(i,j) = \max \begin{cases} 0 \\ F(i-1,j-1) + s(x_i, y_j) \\ F(i-1,j) - d \\ F(i,j-1) - d \end{cases}$$

Final similarity score = 29

Best alignment : G G C T G C A A C T A G C
G G G T A - A G C T T G C

2) Identify dinucleotide CA repeat region and the score in the following sequence: TGGCACACTCACACCACACAGACAGTTA

The sequence can be shown as

TGG[CA][CA]CT[CA][CA]C[CA][CA][CA]GA[CA]GTTA //

Tandem repeat region : CACACCACAC

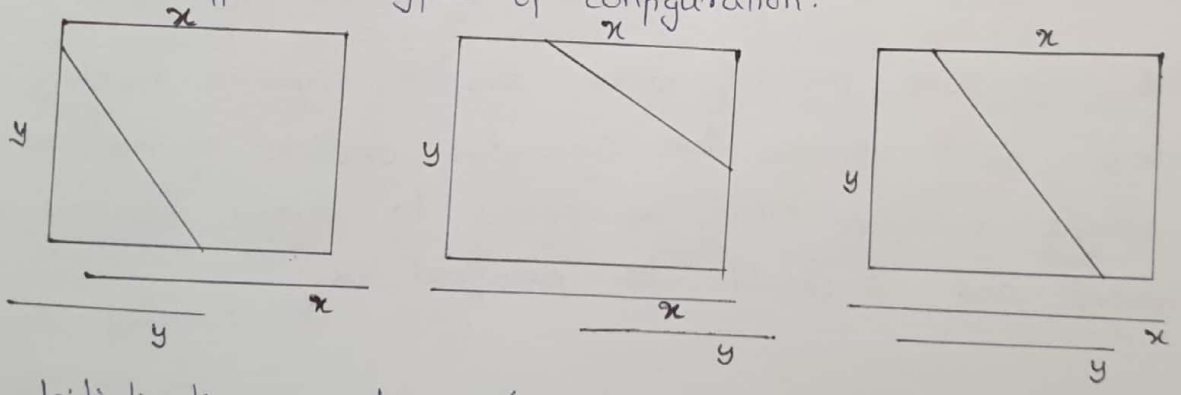
with repeat element CACAC.

Score of CACAC = 20.

3) Overlap sequences are observed when one sequence is contained in another, or the two sequences have some common overlapping regions

These situations come forth when we are comparing fragments of genomic DNA sequence to each other or to large chromosomal sequences, like in sequence assembly

The different types of configuration:



Initialization equations: (same as that for local alignment)

$F(0,0) = 0$

$F(i,0) = 0 \quad \forall \quad i = 1, \ldots, n$

$F(0,j) = 0 \quad \forall \quad j = 1, \ldots, m$

Recurrence relations: (same as that for global alignment)

$$F(i,j) = \max \begin{cases} F(i-1, j-1) + s(x_i, y_j \\ F(i-1, j) - d \\ F(i, j-1) - d \end{cases}$$

Boundary conditions:

Start — $F(0,j)$ or $F(i,0)$ $\quad i = 1, \ldots, n$

End — $F(i,m)$ or $F(n,j)$ $\quad j = 1, \ldots, m$

Traceback conditions:

Traceback starts from f_{max}, where F_{max} is the maximum value on the bottom border (i,m) or on the right border (n,j) and continues till top $(i,0)$ or left $(0,j)$ edge is reached for any $i = 1, \ldots, n$ and $j = 1, \ldots, m$

4) Affine gap score : $\gamma(g) = -d - (g-1)e$

where, d :- gap open penalty

e :- gap extension penalty

g :- number of the consequent gaps

$e < d$ allows long insertions and deletions to be penalized less compared to that obtained by linear gap score.

Affine gap score provides more sensitive sequence matching methods as it assumes that consecutive deletions or insertions are a single mutation event as opposed to multiple insertions or deletions and so, should be penalized less.

5) Time complexity of DP = $O(nm)$ $\quad$ n, m are length of two sequences

Space complexity of DP = $O(nm)$

Time complexity may create problems in database search, where a query sequence of length 'n' is searched in a database of a few GBs in size (approx)

Space complexity would be a problem when comparing complete genomes/chromosomes that are atleast a few MBs long

6) Score matrix constructed as follows:

A log-odds approach is followed where score is proportional to the log of ratio of target frequencies to background frequencies i.e. score matrix for time 't' is given by:

$$S(a,b|t) = \log \frac{P(a|b,t)}{q_a q_b}$$

$P(a|b,t)$:- conditional probability that 'b' is substituted by 'a' in 't'

q_a, q_b :- frequencies of AAs in 'a', 'b' respectively.

First align closely related sequences (>85% identity) and then observe the probability of AA changes and compute the logodds ratio.

Then normalize the matrix to give average change of 1% of all positions to obtain PAM-1 matrix.

To derive scoring matrices for distantly related sequences from data about closely related sequences, extrapolate PAM-1 (through successive iteration of a reference mutation matrix) to get the scoring matrices to any PAM distance as follows: $M_n = (M_1)^n$

M_n — substitution probabilities after n PAMs

M_1 — matrix reflecting 99% sequence conservation and one accepted point mutation (PAM-1) per 100 residues

7) We get more significant matches for protein search

Reasons are:

- DNA made of just 4 characters (A, T, G, C). So, even two unrelated DNA are expected to have 25% similarity (approx.) However, a Protein sequence is composed of around 20 different Amino Acids which improves the comparison sensitivity and hence high similarity would usually imply homology
- Very different DNAs can code for similar protein sequences giving more random matches in the DNA search compared to proteins.
- DNA databases are much larger and grow faster than protein databases and so experience more random hits.
- For DNA, we generally use identity matrices whereas for proteins we employ more sensitive matrices such as PAM and BLOSUM. These usually result in better search results Proteins rarely mutate during evolution. Due to this conservation, searching proteins reveal remote evolutionary relationships

8) PSI BLAST (Position Specific Iterated BLAST):-

PSI-BLAST allows user to build a position-specific scoring matrix derived during the search itself unlike BLAST. Designed to find remote homologues (15-25% identity levels) It is done by constructing scoring matrices by multiple alignment of hits obtained. First we will search the database with the new scoring matrix for every iteration and iterate until convergence is reached. This concept of a tailor-made new scoring matrix to find sequences similar to the query allowing detection of homologues in 15-25% identity levels

BLAST (Basic Local Alignment Search Tool) :—

BLAST is a heuristic algorithm designed to find high scoring local alignments between a query sequence and a target database.

We look for High-scoring Segment Pairs (HSP) by using a scoring matrix to score aligned pairs. Only those pairs which score above a threshold are considered for extension. These HSPs are then extended in both directions to get Maximal Segment Pairs (MSP) until score drops below a threshold drop-off from the maximum score encountered. This gives us the MSP between two sequences.

9)(i) Relative magnitudes of the match score (M) and mismatch score (N) determines the no. of nucleic acid PAMs (Point Accepted Mutations per 100 residues) for which they are most sensitive at finding homologues. Therefore, the reward/penalty ratio should be increased as one observes more divergent sequences. Hence match/mismatch ratio for comparing nucleotide sequences is chosen to be large for highly conserved sequences, while it is small for diverse sequences.

(ii) A match (M)/mismatch (N) ratio of 0.5 (1/-2) is the best for 95% conserved sequences.

	A	C	G	T
A	1	-2	-2	-2
C	-2	1	-2	-2
G	-2	-2	1	-2
T	-2	-2	-2	1

10) Tryptophan and Cytosine are often found at key positions in proteins where they play a critical role. These amino acids have unique chemistries and often play important structural and catalytic roles in proteins. Hence they cannot be substituted easily. So, two aligned Tryptophan get a high score (11) in the substitution matrix while two aligned Leucine get much lower score (4), due to the fact that Leucine residues can often be easily substituted by other amino acids.

11) 65% identity $\Rightarrow$

Target probability of match = .65

Target probability of mismatch = .35

Background probability = .25

According to log-odds approach

$$S(a,b|t) \propto \frac{\log(\text{Target probability})}{\text{Background probability}}$$

$$= \frac{\log P(a|b,t)}{0.25}$$

$$\text{log-odds for match} = \log\left(\frac{0.65}{0.25}\right) = 1.378$$

$$\text{log-odds for mis-match} = \log\left(\frac{0.35}{0.25}\right) = 0.485$$

	A	T	G	C
A	1.378	0.485	0.485	0.485
T	0.485	1.378	0.485	0.485
G	0.485	0.485	1.378	0.485
C	0.485	0.485	0.485	1.378

Normalizing the matrix, we observe that score for match = 17 ; score for mismatch = 6